

AMENDMENTS TO THE SPECIFICATION

Please replace the Abstract with the following amended Abstract:

An automated system for validation, scripting, dissemination and installation of software on information handling systems is disclosed. Operation of the system is triggered by the submission of a software package from a third party. The automated system instantly scans the software package for viruses prior to distributing the software to internal systems in an automated build-to-order manufacturing system. The software package is then processed to verify compliance in accordance with a predefined set of software rules. Appropriate messages are automatically generated upon detection of a compliance failure. The software package is then automatically disassembled and repackaged to integrate it directly into existing management applications within the build-to-order manufacturing system. Thereafter, factory scripts are generated and the software is disseminated to file servers for automatic configuration, execution and validation of factory test images. The method and apparatus of the present invention reduces human error and significantly reduces the time required to configure and execute factory image testing. In addition, it provides centralized audit-trailing of all actions associated with the specific software package received from a third party.

Please replace the paragraph on page 2, beginning on line 6, with the following amended paragraph:

In recent years, there has been an increase in the number of information handling systems that are manufactured based on a “build to order” process that allows a customer to specify specific hardware and software options. In general, prior art “build to order” systems require a large number of disjoint, manual processes that must take place for a manufacturer to successfully validate and transform third-party vendor software packages into “factory installable bits.” This not only increases the probability of human-induced error, but dramatically increases the time required to deliver validated, installable software images to the factory environment. In view of the foregoing, there is a need for an automated system for receiving software and generating validated[[,]] “factory-

installable bits” without direct human interaction. Such a system is provided by the present invention, as described hereinbelow.

Please replace the paragraph on page 5, beginning on line 11, with the following amended paragraph:

Component descriptors are computer readable descriptions of the components of target information handling system 120 which components are defined by the order 110. In a preferred embodiment, the component descriptors are included in a descriptor file called a system descriptor record which is a computer readable file containing a listing of the components, both hardware and software, to be installed onto target information handling system 120. Having read the plurality of component descriptors, database server 140 provides a plurality of software components corresponding to the component descriptors to file server 142 over network ~~connection~~ connections 144. Network connections 144 may be any network connection well-known in the art, such as a local area network, an intranet, or the internet. The information contained in database server 140 is often updated such that the database contains a new factory build environment. The software is then installed on the target information handling system 120. The software installation is controlled by a software installation management server, discussed in greater detail below, that is operable to control the installation of the operating system and other software packages specified by a customer.

Please replace the paragraph on page 5, beginning on line 19, with the following amended paragraph:

Figure 2 is a generalized illustration of an information handling system, such as the target information handling system 120 illustrated in Figure 1. The information handling system includes a ~~processor~~ CPU 202, input/output (I/O) devices 204, such as a display, a keyboard, a mouse, and associated controllers, a hard disk drive 206, and other storage devices 208, such as a floppy disk and drive and other memory devices, and various other subsystems 210, all interconnected via one or more buses 212. The software that is installed according to the versioning methodology is installed onto hard disk drive 206. Alternately, the software may be installed onto any appropriate non-

volatile memory. The non-volatile memory may also store the information relating to which factory build environment was used to install the software. Accessing this information enables a user to have additional systems corresponding to a particular factory build environment to be built.

Please replace the paragraph on page 6, beginning on line 21, with the following amended paragraph:

Figure 3 is an illustration of the components of the automated system 300 for converting, optimizing and disseminating software in accordance with the present invention. A third party 302 delivers a software package 304 to the system via a firewall 306. If the software package 304 is verified and authorized to pass through the firewall, it is received by a software dissemination server 308. The software dissemination server 308 scans the software package for viruses and transfers the file to a compliance validation server 310 which verifies that the software package complies with a predetermined set of rules required by the build-to-order automated manufacturing system. Examples of such rules include, but are not limited to, naming conventions, length of directory path names, “hidden only” attributes, and “read only” attributes. If the compliance validation server 310 determines that the software package 304 contains errors or that it fails to comply with the predetermined rules, an e-mail notice is sent to the third party 302 to notify the third party about the specific errors and non-compliance parameters that were detected. If, however, the compliance validation server 310 determines that the software package is in compliance, then a compliance validation notice is sent to the repack and script regeneration server 312 which then downloads the software package from the software dissemination server 308.

Please replace the paragraph on page 7, beginning on line 14, with the following amended paragraph:

The repack and script regeneration server 312 disassembles the software package 304 and repackages the software to integrate it directly into existing management applications. In this process, the repack and script generation server 312 generates factory scripts for each package to produce “factory installable bits.” These “factory

installable bits” are then transferred to the download server 316. A copy of the software package 304 is also transferred to an archive server 314. The repack and script regeneration server 312 then generates a signal authorizing the script and ~~installation~~ installer validation server 318 to generate appropriate commands to control downloading of software by the target information handling system 120. The results of the installation are monitored by the script and installer validation server 318 and results are communicated to the software dissemination server 308 while the actual software images are downloaded by the download server 316 onto the hard drive or other storage media of the target information handling system 120.

Please replace the paragraph on page 8, beginning on line 10, with the following amended paragraph:

Figure 4 is a flowchart illustration of the processing steps implemented by the system of the present invention. In step 402, a project is defined by various parameters, such as the software part number (SRV) and operating systems (OSes) and the associated test cases for a particular software package. In step 404, a software application is received from a third party. In step 406, the software package received from the third party is scanned for viruses. In step 408, the software package is analyzed to confirm that it complies with a predetermined set of rules to ensure error-free integration into the factory manufacturing system. In step 410, a test is conducted to determine if any compliance errors have been detected. If the result of the test conducted in step 410 indicates that the software package is not compliant, the third party is notified in step 412 of the specific non-compliance factors. If, however, the test conducted in step 410 indicates that the software is compliant, processing proceeds to step 414 where the software package is disassembled, separated, then reassembled into one or more usable packages for integration into the factory manufacturing environment and factory scripts are generated for each package to derive “factory installable bits.” In step 416, the reassembled software package is archived in archive server 314. In step 418, the factory script is validated and installation of the software package is authorized. In step 420, the software images are downloaded to the target information handling system package is installed, and in step 422 the installed software is tested.